UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/905,501	10/01/2007	Emad Moussa Boctor	02240-284171	5737

101887 7590 02/01/2017
The Johns Hopkins University
C/O Venable LLP
P. O. Box 34385
Washington, DC 20043-9998

EXAMINER
EVOY, NICHOLAS LANE

ART UNIT	PAPER NUMBER
3768	

NOTIFICATION DATE	DELIVERY MODE
02/01/2017	ELECTRONIC

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

hjdaley@venable.com
PTOMAIL@VENABLE.COM
cavanhouten@VENABLE.COM

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

Ex parte EMAD MOUSSA BOCTOR, GABOR FICHTINGER,
GREGORY D. HAGER, and HASSAN RIVAZ

Appeal 2015-004875
Application 11/905,501[1]
Technology Center 3700

Before JENNIFER D. BAHR, GEORGE R. HOSKINS, and
FREDERICK C. LANEY, *Administrative Patent Judges*.

LANEY, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF THE CASE

Emad Moussa Boctor et al. (Appellants) appeal under 35 U.S.C. § 134(a) from the Examiner's final decision rejecting claims 5, 6, 10, 11, 26, and 27 under 35 U.S.C. § 102(b) as anticipated by Hall (US 6,508,768 B1, iss. Jan. 21, 2003).[2] We have jurisdiction over this appeal under 35 U.S.C. § 6(b).

We REVERSE.

---

[1] According to Appellants, the real party in interest is The Johns Hopkins University. Appeal Br. 3 (filed October 29, 2014).

[2] Claims 1–4 and 16–25 have been withdrawn. Appeal Br. 15, 17–19 (Claims App.). Claims 7–9 and 12–15 have been canceled. *Id.* at 16, 17 (Claims App.).

INVENTION

Appellants' invention relates to an ultrasound imaging application for measuring tissue elasticity. Spec. ¶ 2.

Claim 5, reproduced below, is independent and illustrative of the claimed invention.

> 5. A method for computing an ultrasound image, comprising:
>
> acquiring first ultrasound data from an ultrasound probe from a region of interest under a first stress state;
>
> sampling the first ultrasound data to provide a first plurality of data samples;
>
> segmenting the first plurality of data samples into a first plurality of discrete elements corresponding to a first ultrasound image;
>
> acquiring second ultrasound data from the ultrasound probe from the region of interest under a second stress state;
>
> sampling the second ultrasound data to provide a second plurality of data samples;
>
> segmenting the second plurality of data samples into a second plurality of discrete elements corresponding to a second ultrasound image;
>
> calculating a degree of similarity of each of said first plurality of discrete elements to a subset of said second plurality of discrete elements and computing a plurality of costs corresponding to each of said first plurality of discrete elements using *a dynamic programming procedure*;
>
> selecting a displacement for each of said first plurality of discrete elements that corresponds to a minimum cost using said dynamic programming procedure so as to provide a computed displacement field; and
>
> computing the ultrasound image using the computed displacement field, wherein said first stress state is different from said second stress state.

Appeal Br. 15–16 (Claims App.) (emphasis added).

ANALYSIS

Appellants raise a single issue regarding the Examiner's finding that Hall anticipates the claimed invention, which focuses on whether Hall discloses the use of a "dynamic programming procedure." Appeal Br. 8–13. The Examiner finds Hall's disclosure of "comparing the two kernels to minimize a 'cost or energy function' . . . and 'dynamically adjusting' the search region within a programmed procedure . . . would be considered a dynamic programming procedure." Final Act. 3 (emphasis omitted). Supporting that finding, the Examiner concludes,

> one of ordinary skill in the art would recognize that the term "dynamic programming procedure" can encompass any process implemented on, or in any way uses, a computer readable medium (i.e. a "programming procedure", see http://www.merriam-webster.com/dictionary/program) which is in some way, shape, or form, "dynamic" (i.e. marked by change or continuous activity, see http://www.merriam-webster.com/dictionary/dynamic).

Id. at 7–8.

Appellants object to the Examiner's interpretation as being unreasonably broad because the term "dynamic processing" has an accepted meaning by those of skill in the art, which Appellants contend "would have been expected to have had at least an undergraduate level of understanding of computer vision." Appeal Br. 9–10. Appellants submit a skilled artisan understands the term "dynamic programming" refers to "a standard algorithm for optimizing a recursively defined objective function, that is, an objective where an optimal sequence of decisions can be expressed as a combination of an optimal solution for a subsequence together with an optimal solution for the next step." *Id.* at 10 (quoting Hager Decl.[3] ¶ 7). Under this interpretation, Appellants contend the evidence the Examiner cites fails to support the finding Hall discloses using a "dynamic processing procedure," as required by each of the claims, either directly or indirectly. *Id.* at 10–11. As a result, the resolution of this appeal rests on the proper interpretation of the term "dynamic programming procedure."

The Examiner does not challenge Appellants' representation of the level of skill of a skilled artisan. The Examiner agrees "a 'standard algorithm for optimizing a recursively defined objective function'" is an accepted meaning for that term, but concludes it is not the broadest reasonable interpretation. Ans. 8. In addition, the Examiner agrees the Specification, at paragraphs 58, 59, 63, 64, 67, and 70, provides examples consistent with Appellants' proposed interpretation, but asserts, "one of ordinary skill in the art would not recognize these examples as *required features* under the plain and ordinary meaning of the term." *Id.* at 9. The Examiner asserts, "the term is used in other references in various contexts which are not limited to the definition provided by [Appellants]." *Id.* at 8. As support, the Examiner cites "Chapter 20 of *Operations Research Models and Methods* by Paul A. Jensen and Jonathan F. Bard (2002)" (hereinafter "Jensen")[4] and represents the authors as explicitly noting, "that dynamic programming is 'much more general than linear programming' and that 'there is no common data structure that unifies all [dynamic programming]'"

---

[3] Declaration of Co-Inventor Gregory D. Hager (executed and filed with the USPTO on Jan. 13, 2014).

[4] The Examiner indicates that the Jensen reference can be accessed at http://www.me.utexas.edu/~jensen/ORMM/sugglements/units/dp methods/.

and as teaching, "that dynamic programming '[m]odels are problem specific, and for the most part, are represented by recursive formulas rather than algebraic expressions.'" *Id.* at 8–9.

"The broadest-construction rubric . . . does not give the PTO an unfettered license to interpret claims to embrace anything remotely related to the claimed invention." *In re Suitco Surface, Inc.*, 603 F.3d 1255, 1260 (Fed. Cir. 2010).

> Since it would be unreasonable for the PTO to ignore any interpretive guidance afforded by the applicant's written description . . . the PTO applies to the verbiage of the proposed claims the broadest reasonable meaning of the words in their ordinary usage as they would be understood by one of ordinary skill in the art, taking into account whatever enlightenment by way of definitions or otherwise that may be afforded by the written description contained in the applicant's specification.

In re Morris, 127 F.3d 1048, 1054 (Fed. Cir. 1997). In this case, the Examiner's reliance exclusively on the collective meanings of "dynamic" and "programming," as defined by a general purpose dictionary (i.e., Merriam-Webster's), to establish the broadest reasonable interpretation of the term "dynamic programming" led to an unreasonably broad construction.

Given the Examiner does not dispute Dr. Hager's testimony that a skilled artisan in this field of art recognizes the term "dynamic programming" to mean "a standard algorithm for optimizing a recursively defined objective function," and agrees the Specification uses that term in a manner consistent with this understanding, the Examiner errs by disregarding these facts in favor of a meaning derived from combining the general definitions of the individual words comprising the disputed term. We note, consistent with Appellants' interpretation, two dictionaries more closely related to the field of art define "dynamic programming" as, "a procedure for optimizing a multi-stage problem solution, in which a number of decisions are available at each stage of the process." *Dynamic Programming*, The Authoritative Dictionary of IEEE Standard Terms (7th ed. 2000); *Dynamic Programming*, IBM Dictionary of Computing (1994).

The Examiner's reliance on Jensen is also misplaced because it is actually consistent with the reference Dr. Hager cites to support his testimony and Appellants' construction. In particular, at pages 15–16, Exhibit A of Dr. Hager's declaration, a survey by Pedro F. Felzenszwalb and Ramin Zabih titled "Dynamic Programming and Graph Algorithms in Computer Vision," states:

> Dynamic programming . . . is a powerful general technique for developing efficient discrete optimization algorithms. . . . The basic idea of dynamic programming is to decompose a problem into a set of subproblems, such that (A) given a solution to the subproblems, we can quickly compute a solution to the original problem, and (B) the subproblems can be efficiently solved recursively in terms of each other. An important aspect of dynamic programming is that the solution of a single subproblem is often used multiple times, for solving several larger subproblems. . . . Similar to shortest paths algorithms, dynamic programming relies on an optimal substructure property. This makes it possible to solve a subproblem using solutions of smaller subproblems. . . . The dynamic programming algorithm iterates over table entries and computes a value for the current entry using values of entries that it has already computed. Often there is a simple recursive equation which defines the value of an entry in terms of values of other entries, but sometimes determining a value involves a more complex computation.

Similarly, Jensen states, at Chapter 20, pages 1–2,

> Although all dynamic programming (DP) problems have a similar structure, the computational procedures used to find solutions are often quite different. Unlike linear and integer

> programming where standard conventions are used to describe parameters and coefficients, there is no common data structure that unifies all DPs. Models are problem specific, and for the most part, are represented by recursive formulas rather than algebraic expressions. . . . Dynamic programming is, however, much more general than linear programming with respect to the class of problems it can handle. . . . The central requirement of a dynamic programming model is that the optimal sequence of decisions from any given state be independent of the sequence that leads up to that state. All computational procedures are based on this requirement known as the *principle of optimality* so the model must be formulated accordingly.

In both references, dynamic programming is described as "a standard algorithm for optimizing a recursively defined objective function." It is also worth noting that Jensen clearly indicates a skilled artisan recognized "dynamic programming" as being "unlike linear and integer programming where standard conventions are used to describe parameters and coefficients"; nevertheless, the Examiner's broad construction makes no distinction. For the foregoing reasons, and because the Examiner has not made any factual findings that Hall discloses the use of "dynamic programming" under the proper construction, we do not sustain the Examiner's finding that Hall anticipates claims 5, 6, 10, 11, 26, and 27.

DECISION

The Examiner's rejections of claims 5, 6, 10, 11, 26, and 27 are reversed.

REVERSED